

NightRiderV1(残像方式)

概要

『ナイトライダー(Knight Rider)』は、1982年9月から1986年8月にかけて、米国で放映された特撮テレビドラマです。そこに登場するK.I.T.Tという高性能人工知能を搭載した「ナイト2000(トランザム)」のボンネットには、残像を残しながら、左右に流れる赤いランプが搭載されていました。そこでこれを再



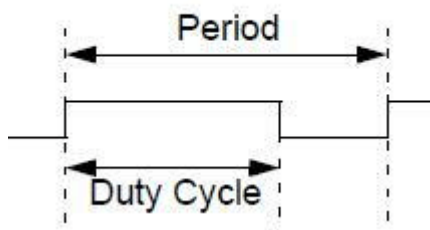
現(?)してみようと思ひ立ち製作してみました。

動作原理

単にLEDが左右に流れるだけでは、面白くありませんので、残像を残しながら流れるように工夫してみました。つまり、先頭のLEDは明るく、後に続くLEDは徐々に暗くしていきます。

LEDの明るさの制御は、PWM(Pulse Width Modulation)方式としました。PIC16F88にはPWMモジュールが搭載されていますが、1個だけなので利用はあきらめ、ソフトウェアで10個のPWMを実現することにしました。

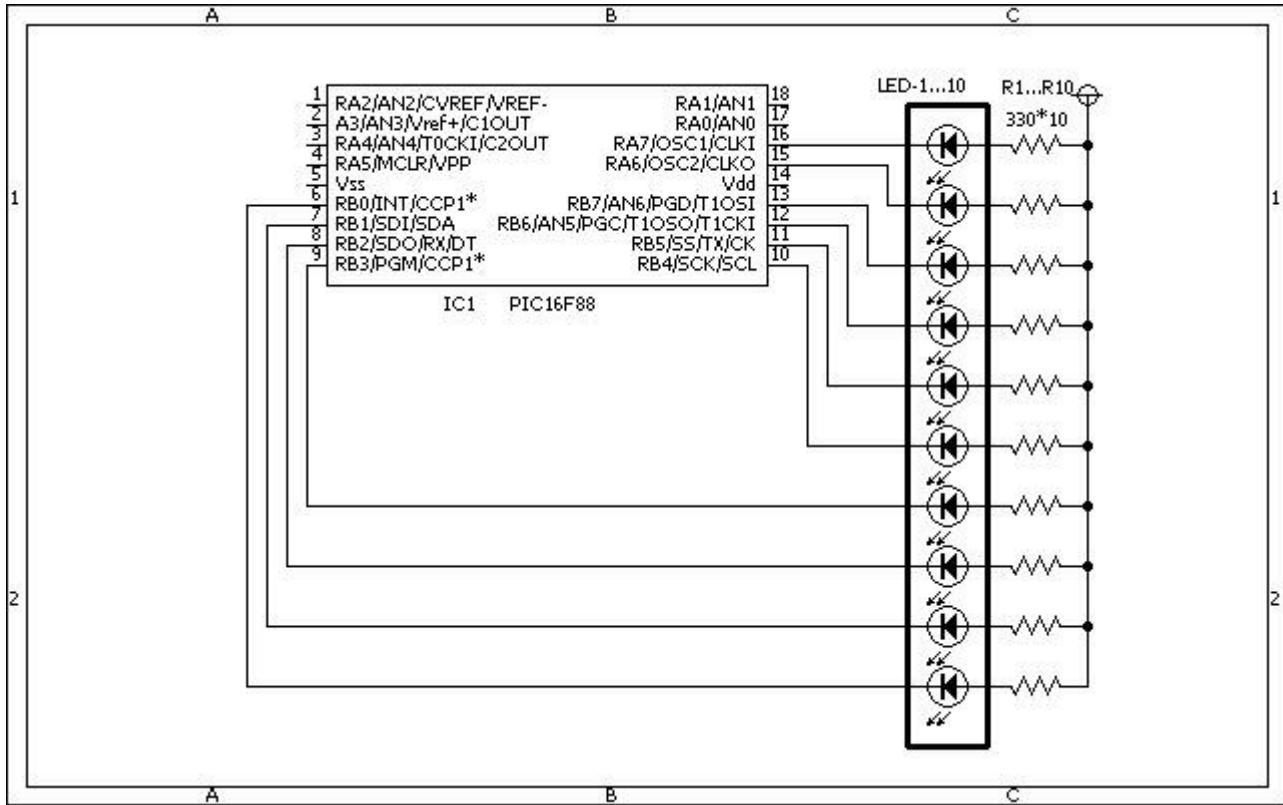
PWMを使った制御では、周期(period)を設定し、デューティ比(duty cycle)を増減させることによ



りLEDの明るさを可変します。

今回は、明るさを10段階としました。 <周期> TIMER0の割り込みと割り込み回数を使用して、約33msecとします。 (1秒 ÷ 8MHz) × 4 (Fosc/4) × 256 (TMR0) × 255 (割り込み回数) <デューティ比> 100%(255)、50%(128)、25%(64)、13%(32)、6%(16)、3%(8)、1.5%(4)、0.8%(2)、0.4%(1)、0%(0)

回路図



ソースコード

NightRiderV1.c

```

//*****
*
/*
□□□□□□□□□□□□□□□□残像機能搭載) >
*/
//*****
*

#define      ON      0
#define      OFF     1

#define      LED1    PORTB.F0
#define      LED2    PORTB.F1
#define      LED3    PORTB.F2
#define      LED4    PORTB.F3
#define      LED5    PORTB.F4
#define      LED6    PORTB.F5
#define      LED7    PORTB.F6
#define      LED8    PORTB.F7
#define      LED9    PORTA.F6
#define      LED10   PORTA.F7

//*****

```

```
*  
  
void SwitchONcheck()  
{  
    while (Button(&PORTA, 2, 1, 0) == 0)  
        ;  
    while (Button(&PORTA, 2, 1, 1) == 0)  
        ;  
}  
  
//*****  
*  
  
static unsigned short pwm_cnt, led_cnt[10];  
  
void interrupt()  
{  
    if (INTCON.T0IF == 1) { // 約0.128msec周期  
        INTCON.T0IF = 0;  
        //  
        LED1 = (pwm_cnt <= led_cnt[0]) ? ON : OFF;  
        LED2 = (pwm_cnt <= led_cnt[1]) ? ON : OFF;  
        LED3 = (pwm_cnt <= led_cnt[2]) ? ON : OFF;  
        LED4 = (pwm_cnt <= led_cnt[3]) ? ON : OFF;  
        LED5 = (pwm_cnt <= led_cnt[4]) ? ON : OFF;  
        LED6 = (pwm_cnt <= led_cnt[5]) ? ON : OFF;  
        LED7 = (pwm_cnt <= led_cnt[6]) ? ON : OFF;  
        LED8 = (pwm_cnt <= led_cnt[7]) ? ON : OFF;  
        LED9 = (pwm_cnt <= led_cnt[8]) ? ON : OFF;  
        LED10 = (pwm_cnt <= led_cnt[9]) ? ON : OFF;  
        //  
        if (pwm_cnt < 255)  
            pwm_cnt++;  
        else  
            pwm_cnt = 1;  
    }  
}  
  
//*****  
*  
  
unsigned short Luminance[10] = {255, 128, 64, 32, 16, 8, 4, 2, 1,  
0};  
//unsigned short Luminance[10] = {255, 64, 16, 4, 1, 0, 0, 0, 0,  
0};  
  
void main()  
{  
    unsigned short cnt;  
    //  
    CMCON = 0b00000111;  
}
```

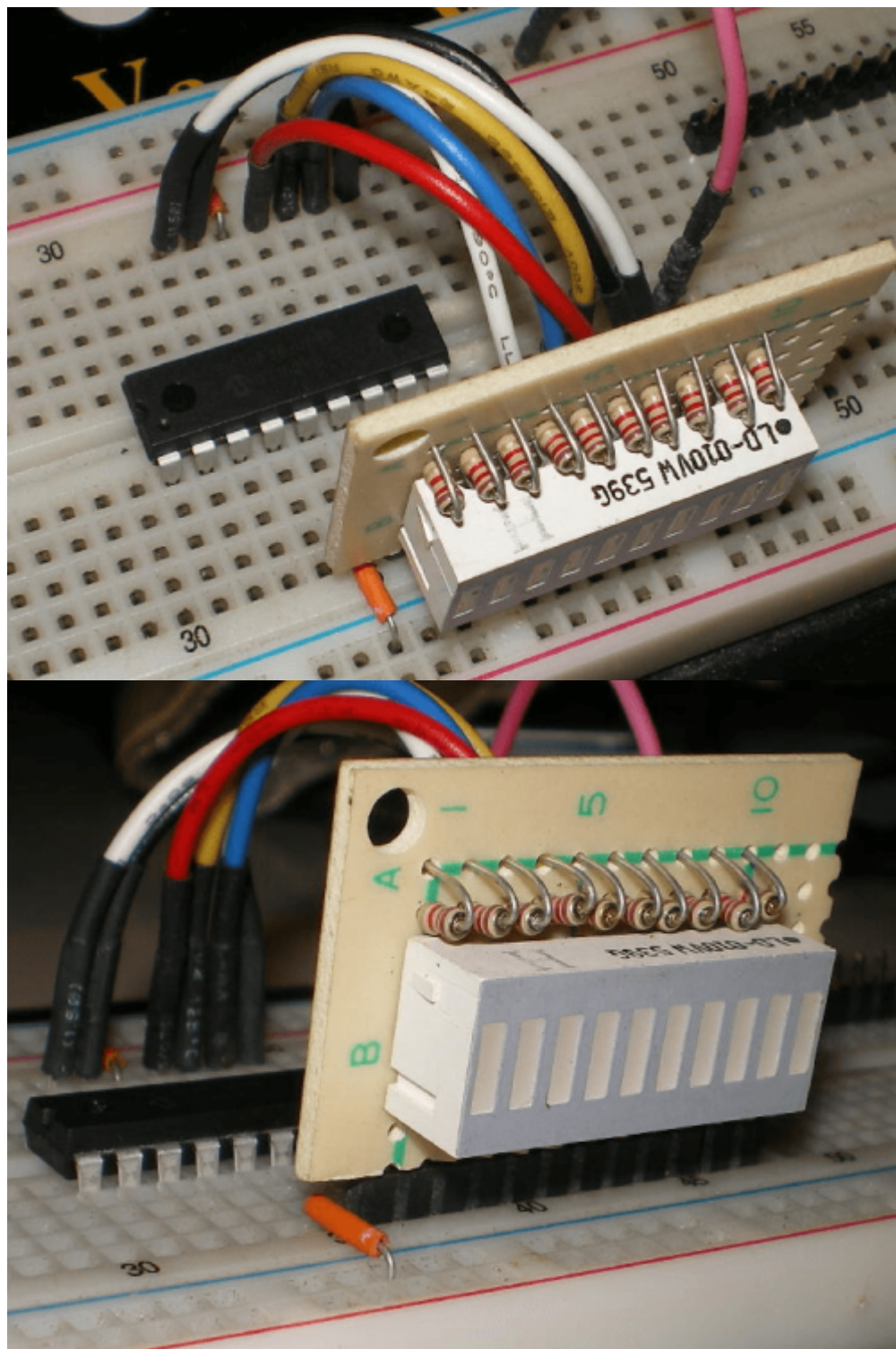
```
ANSEL = 0b00000000;
TRISA = 0b00111100;
TRISB = 0b00000000;
OSCCON = 0b01110000;
// TIMER0の設定
INTCON.T0IE = 1;
INTCON.T0IF = 0;
OPTION_REG.T0CS = 0;
OPTION_REG.PSA = 1;
OPTION_REG.PS0 = 0;
OPTION_REG.PS1 = 0;
OPTION_REG.PS2 = 0;
TMR0 = 0;
//
pwm_cnt = 1;
// 割り込みを許可する。
INTCON.PEIE = 1;
INTCON.GIE = 1;
//
for (cnt = 0; cnt < 10; cnt++) {
    led_cnt[cnt] = Luminance[cnt];
}
Delay_ms(1000);
//
for (cnt = 0; cnt < 10; cnt++) {
    led_cnt[cnt] = Luminance[9 - cnt];
}
Delay_ms(1000);
//
while (1) {
    for (cnt = 0; cnt < 10; cnt++) {
        led_cnt[9] = led_cnt[8];
        led_cnt[8] = led_cnt[7];
        led_cnt[7] = led_cnt[6];
        led_cnt[6] = led_cnt[5];
        led_cnt[5] = led_cnt[4];
        led_cnt[4] = led_cnt[3];
        led_cnt[3] = led_cnt[2];
        led_cnt[2] = led_cnt[1];
        led_cnt[1] = led_cnt[0];
        led_cnt[0] = Luminance[cnt];
        Delay_ms(30);
    }
    for (cnt = 0; cnt < 10; cnt++) {
        led_cnt[9] = led_cnt[8];
        led_cnt[8] = led_cnt[7];
        led_cnt[7] = led_cnt[6];
        led_cnt[6] = led_cnt[5];
        led_cnt[5] = led_cnt[4];
        led_cnt[4] = led_cnt[3];
        led_cnt[3] = led_cnt[2];
    }
}
```

```
        led_cnt[2] = led_cnt[1];
        led_cnt[1] = led_cnt[0];
        Delay_ms(30);
    }
    for (cnt = 0; cnt < 10; cnt++) {
        led_cnt[0] = led_cnt[1];
        led_cnt[1] = led_cnt[2];
        led_cnt[2] = led_cnt[3];
        led_cnt[3] = led_cnt[4];
        led_cnt[4] = led_cnt[5];
        led_cnt[5] = led_cnt[6];
        led_cnt[6] = led_cnt[7];
        led_cnt[7] = led_cnt[8];
        led_cnt[8] = led_cnt[9];
        led_cnt[9] = Luminance[cnt];
        Delay_ms(30);
    }
    for (cnt = 0; cnt < 10; cnt++) {
        led_cnt[0] = led_cnt[1];
        led_cnt[1] = led_cnt[2];
        led_cnt[2] = led_cnt[3];
        led_cnt[3] = led_cnt[4];
        led_cnt[4] = led_cnt[5];
        led_cnt[5] = led_cnt[6];
        led_cnt[6] = led_cnt[7];
        led_cnt[7] = led_cnt[8];
        led_cnt[8] = led_cnt[9];
        Delay_ms(30);
    }
}

//*****
*
```

動作確認

LEDの抵抗値は、好みの明るさに合わせて調整してみてください。



左から右へ残像を残しながら光が流れます。 デジカメなので今一つですが、目視では結構綺麗に見えま



す。

右から左へ残像を残しながら光が流れます。



如何ですか? これから年末(クリスマス)にかけて、光物が宜しいかと。。。 😊

著作権表示 **copyright notice**

このページは稲崎様の閉鎖したHPのコピーで、著作権は稲崎様にあります。 [詳細](#) This page is a copy of Mr. Inasaki's closed website, and the copyright is held by him. [Details](#)

From:

<http://www.deepsky.jp/wiki/> - うごくといいな

Permanent link:

<http://www.deepsky.jp/wiki/doku.php?id=elechobby:picdic:pic16f88:98>

Last update: **2025/10/17 14:29**

